

Lógica Computacional

Aula Teórica 20: Forma Normal Prenex

António Ravara

Departamento de Informática

9 de Maio de 2011



Formas normais na primeira ordem

Objectivo

Determinar a validade de raciocínios (ou de fórmulas) semânticamente, mas de forma eficaz e automática.

Meio: algoritmo

Define-se uma função que recebe uma fórmula e devolve a sua natureza (possível, contraditória ou válida).

Abordagens computacionais

Há vários algoritmos, de acordo com formas especiais em que as fórmulas podem estar. Uns são mais eficazes que outros. Vamos ver o mais eficaz: a *resolução*.

Forma Normal Prenex

Ideia

Os quantificadores estão todos “na cabeça” da fórmula.

Exemplos

- $Q(x) \rightarrow P(x, y);$
- $\forall x \exists y f(x) = y;$
- $\forall x \exists y P(x, y, z);$
- $\exists x (f(x) = y \wedge P(x, y, z)).$

Contra-Exemplos

- $Q(x) \rightarrow \exists y P(x, y);$
- $\forall x \exists y f(x) = y \wedge \forall x \exists y P(x, y, z);$
- $f(x) = y \wedge \forall x \exists y P(x, y, z);$
- $\neg \exists x (f(x) = y \wedge P(x, y, z)).$

Forma Normal Prenex

Definição

Uma fórmula φ da linguagem de primeira ordem está na Forma Normal Prenex ou FNP (e escreve-se $\text{FNP}(\varphi)$), se

$$\varphi = Q_1 x_1 \dots Q_n x_n \psi$$

sendo:

- Cada Q_i um quantificador, com $1 \leq i \leq n$ e $n \geq 0$;
- ψ uma fórmula de primeira ordem sem quantificadores.

Se ψ está na forma normal conjuntiva (aplicando o conceito a fórmulas de primeira ordem), então φ diz-se uma forma normal conjuntiva prenex ou FNCP (e escreve-se $\text{FNCP}(\varphi)$).

Qualquer fórmula é equivalente a uma FNCP

Lema das Formas Normais

Para qualquer fórmula de primeira ordem φ existe uma fórmula de primeira ordem ψ tal que $\varphi \equiv \psi$ e $\text{FNP}(\psi)$.

Prova

Mostra-se por indução estrutural.

Casos base: as fórmulas atômicas já estão na FNP.

Passo. Analisa-se por casos a forma de φ . Mostra-se a prova de dois casos tipo; os restantes têm prova semelhante.

Seja $\varphi \stackrel{\text{def}}{=} \exists x \phi$. Por hipótese de indução existe $\psi \equiv \phi$ tal que $\text{FNP}(\psi)$, logo $\varphi \equiv \exists y \psi$ para alguma variável y e $\text{FNP}(\varphi)$.

Resultado central

Lema das Formas Normais

Para qualquer fórmula de primeira ordem φ existe uma fórmula de primeira ordem ψ tal que $\varphi \equiv \psi$ e $\text{FNP}(\psi)$.

Prova

Seja agora $\varphi \stackrel{\text{def}}{=} \varphi_1 \wedge \varphi_2$.

Por hipótese de indução existem ψ_1, ψ_2 tal que $\varphi_1 \equiv \psi_1$ e $\text{FNP}(\psi_1)$, e $\varphi_2 \equiv \psi_2$ e $\text{FNP}(\psi_2)$. Logo, sendo $n, m \geq 0$, cada Q_i e R_j quantificadores, com $1 \leq i \leq n$ e $1 \leq j \leq m$, e ψ'_1, ψ'_2 fórmulas de primeira ordem sem quantificadores, tem-se que

$$\psi_1 = Q_1 x_1 \dots Q_n x_n \psi'_1$$

$$\psi_2 = R_1 x_1 \dots R_m x_m \psi'_2$$

Lema das Formas Normais

Continuação da prova: caso $\varphi \stackrel{\text{def}}{=} \varphi_1 \wedge \varphi_2$

Note-se que não se tem que $\text{FNP}(\psi_1 \wedge \psi_2)$. Como transformar para FNP?

Provou-se já que se $x \notin \text{VL}(\psi)$ então $\forall x \varphi \wedge \psi \equiv \forall x (\varphi \wedge \psi)$ e $\exists x \varphi \wedge \psi \equiv \exists x (\varphi \wedge \psi)$.

Facilmente se generaliza este resultado, obtendo-se

$Q_1 x_1 \dots Q_n x_n \varphi \wedge \psi \equiv Q_1 x_1 \dots Q_n x_n (\varphi \wedge \psi)$, se $x_1, \dots, x_n \notin \text{VL}(\psi)$, sendo $Q_1, \dots, Q_n$ quantificadores.

Também se mostrou já a congruência- α , ou seja,

$\forall x \varphi \equiv \forall y \varphi\{y/x\}$ e que $\exists x \varphi \equiv \exists y \varphi\{y/x\}$.

Com estas duas leis mostra-se que existe $\psi \equiv \psi_1 \wedge \psi_2$ tal que $\text{FNP}(\psi)$.

Lema das Formas Normais

Continuação da prova: caso $\varphi \stackrel{\text{def}}{=} \varphi_1 \wedge \varphi_2$

Tem-se então que $\text{FNP}(\psi_1)$ e $\text{FNP}(\psi_2)$ tal que $\varphi_1 \equiv \psi_1$ e $\varphi_2 \equiv \psi_2$, e $\varphi \equiv \psi_1 \wedge \psi_2 = Q_1x_1 \dots Q_nx_n \psi'_1 \wedge R_1x_1 \dots R_mx_m \psi'_2$.

Tomam-se agora variáveis novas (i.e., não usadas nas fórmulas ψ_1 e ψ_2) todas diferentes entre si, $y_1, \dots, y_n, z_1, \dots, z_m$. Seja $\psi''_1 = \psi'_1\{y_1, \dots, y_n/x_1, \dots, x_n\}$ e $\psi''_2 = \psi'_2\{z_1, \dots, z_m/x_1, \dots, x_m\}$

Aplicando repetidamente a congruência- α e duas vezes a lei do âmbito dos quantificadores, usando a comutatividade antes da segunda aplicação, obtém-se uma fórmula em FNP.

$$\begin{aligned}\psi_1 \wedge \psi_2 &= Q_1x_1 \dots Q_nx_n \psi'_1 \wedge R_1x_1 \dots R_mx_m \psi'_2 \\ &\equiv Q_1y_1 \dots Q_ny_n \psi''_1 \wedge R_1z_1 \dots R_mz_m \psi''_2 \\ &\equiv Q_1y_1 \dots Q_ny_n R_1z_1 \dots R_mz_m (\psi''_1 \wedge \psi''_2)\end{aligned}$$

Conversão de uma fórmula

Seja $\varphi = \forall x (P(x) \wedge \exists y \neg R(y, z)) \rightarrow \exists x \forall y Q(x, y, z)$

Faz-se a conversão “manual” de φ para FNP.

$$\begin{aligned}
 \varphi &\equiv \neg \forall x (P(x) \wedge \exists y \neg R(y, z)) \vee \exists x \forall y Q(x, y, z) \\
 &\equiv \exists x \neg (P(x) \wedge \exists y \neg R(y, z)) \vee \exists x \forall y Q(x, y, z) \\
 &\equiv \exists x (\neg P(x) \vee \neg \exists y \neg R(y, z)) \vee \exists x \forall y Q(x, y, z) \\
 &\equiv \exists x (\neg P(x) \vee \forall y \neg \neg R(y, z)) \vee \exists x \forall y Q(x, y, z) \\
 &\equiv \exists x (\forall y (\neg P(x) \vee R(y, z))) \vee \exists x \forall y Q(x, y, z) \\
 &\equiv \exists x \forall y (\neg P(x) \vee R(y, z)) \vee \exists u \forall v Q(u, v, z) \\
 &\equiv \exists x \forall y ((\neg P(x) \vee R(y, z)) \vee \exists u \forall v Q(u, v, z)) \\
 &\equiv \exists x \forall y \exists u \forall v ((\neg P(x) \vee R(y, z)) \vee Q(u, v, z))
 \end{aligned}$$

É possível implementar este procedimento.

Transformação de fórmulas para FNP

Objectivo

Definir um algoritmo para transformar qualquer fórmula de Lógica de Primeira Ordem para a Forma Normal Prenex (FNP).

Conjuntos de fórmulas de primeira ordem

- Seja E_{Σ}^X o conjunto das fórmulas de primeira ordem que se obtém considerando o conectivo de negação ($\neg$) primitivo.
- Seja $G_{\Sigma}^X \subseteq E_{\Sigma}^X$ o conjunto das fórmulas de primeira ordem com negação que não contêm o conectivo de implicação ($\rightarrow$).

Algoritmos auxiliares

Atenção: nas definições das funções recursivas que se apresentam a seguir, assume-se que a análise dos casos se faz sequencialmente (de cima para baixo), sendo cada caso testado apenas se as condições dos anteriores são falsas.

Eliminação da implicação

Seja $\text{ImplFree}: E_{\Sigma}^X \rightarrow G_{\Sigma}^X$ a seguinte função.

$$\text{ImplFree}(\varphi) = \begin{cases} \neg \text{ImplFree}(\varphi_1), & \text{se } \varphi = \neg \varphi_1 \\ \text{ImplFree}(\varphi_1) \vee \text{ImplFree}(\varphi_2), & \text{se } \varphi = \varphi_1 \vee \varphi_2 \\ \text{ImplFree}(\varphi_1) \wedge \text{ImplFree}(\varphi_2), & \text{se } \varphi = \varphi_1 \wedge \varphi_2 \\ \neg \text{ImplFree}(\varphi_1) \vee \text{ImplFree}(\varphi_2), & \text{se } \varphi = \varphi_1 \rightarrow \varphi_2 \\ \forall x \text{ ImplFree}(\varphi_1), & \text{se } \varphi = \forall x \varphi_1 \\ \exists x \text{ ImplFree}(\varphi_1), & \text{se } \varphi = \exists x \varphi_1 \\ \varphi, & \text{caso contrário} \end{cases}$$

Algoritmos auxiliares

Eliminação das duplas negações

Seja NNFC: $G_{\Sigma}^X \rightarrow G_{\Sigma}^X$ a seguinte função.

$$\text{NNFC}(\varphi) = \begin{cases} \text{NNFC}(\varphi_1), & \text{se } \varphi = \neg\neg\varphi_1 \\ \text{NNFC}(\neg\varphi_1) \vee \text{NNFC}(\neg\varphi_2), & \text{se } \varphi = \neg(\varphi_1 \wedge \varphi_2) \\ \text{NNFC}(\neg\varphi_1) \wedge \text{NNFC}(\neg\varphi_2), & \text{se } \varphi = \neg(\varphi_1 \vee \varphi_2) \\ \text{NNFC}(\varphi_1) \vee \text{NNFC}(\varphi_2), & \text{se } \varphi = \varphi_1 \vee \varphi_2 \\ \text{NNFC}(\varphi_1) \wedge \text{NNFC}(\varphi_2), & \text{se } \varphi = \varphi_1 \wedge \varphi_2 \\ \exists x \text{ NNFC}(\neg\varphi_1), & \text{se } \varphi = \neg\forall x \varphi_1 \\ \forall x \text{ NNFC}(\neg\varphi_1), & \text{se } \varphi = \neg\exists x \varphi_1 \\ \forall x \text{ NNFC}(\varphi_1), & \text{se } \varphi = \forall x \varphi_1 \\ \exists x \text{ NNFC}(\varphi_1), & \text{se } \varphi = \exists x \varphi_1 \\ \varphi, & \text{caso contrário} \end{cases}$$

Algoritmos auxiliares

Alargamento do âmbito dos quantificadores

Seja $* \in \{\vee, \wedge\}$, $Q \in \{\forall, \exists\}$, $x_0 \notin V(\varphi)$.

Scope: $G_\Sigma^X \rightarrow G_\Sigma^X$ é a seguinte função.

$$\text{Scope}(\varphi) = \begin{cases} Qx \text{ Scope}(\varphi_1), & \text{se } \varphi = Qx \varphi_1 \\ Qx_0 \text{ Scope}(\varphi_1 \{x_0/x\} * \varphi_2), & \text{se } \varphi = Qx \varphi_1 * \varphi_2 \\ & \text{ou } \varphi = \varphi_2 * Qx \varphi_1 \\ \text{Scope}(\text{Scope}(\varphi_1) * \text{Scope}(\varphi_2)), & \text{se } \varphi = \varphi_1 * \varphi_2 \\ & \text{e ambas têm quant.} \\ \text{Scope}(\text{Scope}(\varphi_1) * \varphi_2), & \text{se } \varphi = \varphi_1 * \varphi_2 \\ & \text{e } \varphi_2 \text{ não tem quant.} \\ \text{Scope}(\varphi_1 * \text{Scope}(\varphi_2)), & \text{se } \varphi = \varphi_1 * \varphi_2 \\ & \text{e } \varphi_1 \text{ não tem quant.} \\ \varphi, & \text{caso contrário} \end{cases}$$

Algoritmo $\mathcal{P}$ de transformação para FNP

Algoritmo $\mathcal{P}$

Seja $\mathcal{P} : E_{\Sigma}^X \rightarrow G_{\Sigma}^X$ a seguinte função.

$$\mathcal{P}(\varphi) = \text{Scope}(\text{NNFC}(\text{ImplFree}(\varphi)))$$

Forma Normal da Negação

Uma fórmula $\varphi \in G_{\Sigma}^X$ diz-se que está na forma normal da negação (e escreve-se $\text{FNN}(\varphi)$), se só as suas subfórmulas que são fórmulas atómicas estão negadas.

Lemas de optimização

- 1 Se $\varphi \in G_{\Sigma}^X$ então $\text{ImplFree}(\varphi) = \varphi$.
- 2 Se $\text{FNN}(\varphi)$ então $\text{NNFC}(\varphi) = \varphi$.
- 3 Se $\text{FNP}(\varphi)$ então $\text{Scope}(\varphi) = \varphi$.

Correcção do algoritmo $\mathcal{P}$

Teorema: conversão de fórmulas de primeira ordem para para FNP

Dada $\varphi \in E_{\Sigma}^X$, a fórmula $\psi = \mathcal{P}(\varphi)$ é tal que $\psi \in G_{\Sigma}^X$, $\varphi \equiv \psi$ e $\text{FNP}(\psi)$.

Assuma-se que o algoritmo CNFC definido para fórmulas proposicionais se aplica também a fórmulas de primeira ordem sem quantificadores.

Lema: conversão para FNCP

Dada uma fórmula $\varphi = Q_1x_1 \dots Q_nx_n \psi$ da linguagem de primeira ordem tal que $\text{FNP}(\varphi)$, obtém-se uma fórmula equivalente ϕ tal que $\text{FNCP}(\phi)$ fazendo $\phi = Q_1x_1 \dots Q_nx_n \text{CNFC}(\psi)$.

Prova do Teorema da correcção do algoritmo $\mathcal{P}$

Lemas auxiliares

- Dada $\varphi \in E_{\Sigma}^X$, a fórmula $\psi = \text{ImplFree}(\varphi)$ é tal que $\psi \in G_{\Sigma}^X$ e $\varphi \equiv \psi$.
- Dada $\varphi \in G_{\Sigma}^X$, a fórmula $\psi = \text{NNFC}(\varphi)$ é tal que $\psi \in G_{\Sigma}^X$, $\varphi \equiv \psi$ e $\text{FNN}(\psi)$.
- Dadas $\varphi \in G_{\Sigma}^X$ tal que $\text{FNN}(\varphi)$ a fórmula $\psi = \text{Scope}(\varphi)$ é tal que $\psi \in G_{\Sigma}^X$, $\varphi \equiv \psi$ e $\text{FNP}(\psi)$.

Exemplo

Seja $\varphi = \forall x \exists y P(x, y, z) \rightarrow \exists x \forall y \neg Q(x, y, z)$

Vai-se calcular $\mathcal{P}(\varphi) = \text{Scope}(\text{NNFC}(\text{ImplFree}(\varphi)))$.

$$\begin{aligned} \text{ImplFree}(\varphi) &= \neg(\text{ImplFree}(\forall x \exists y P(x, y, z))) \\ &\quad \vee \text{ImplFree}(\exists x \forall y \neg Q(x, y, z)) \\ &= \neg \forall x \exists y P(x, y, z) \vee \exists x \forall y \neg Q(x, y, z) = \phi \end{aligned}$$

$$\begin{aligned} \text{NNFC}(\phi) &= \text{NNFC}(\neg \forall x \exists y P(x, y, z)) \vee \text{NNFC}(\exists x \forall y \neg Q(x, y, z)) \\ &= \exists x \text{NNFC}(\neg \exists y P(x, y, z)) \vee \exists x \forall y \neg Q(x, y, z) \\ &= \exists x \forall y \text{NNFC}(\neg P(x, y, z)) \vee \exists x \forall y \neg Q(x, y, z) \\ &= \exists x \forall y \neg P(x, y, z) \vee \exists x \forall y \neg Q(x, y, z) = \gamma \end{aligned}$$

Exemplo

Seja $\gamma = \exists x \forall y \neg P(x, y, z) \vee \exists x \forall y \neg Q(x, y, z)$

$$\begin{aligned}\text{Scope}(\gamma) &= \exists x_0 \forall y_0 \text{Scope}(\neg P(x_0, y_0, z) \vee \exists x \forall y \neg Q(x, y, z)) \\ &= \exists x_0 \forall y_0 \exists x_1 \forall y_1 \text{Scope}(\neg P(x_0, y_0, z) \vee \neg Q(x_1, y_1, z)) \\ &= \exists x_0 \forall y_0 \exists x_1 \forall y_1 (\neg P(x_0, y_0, z) \vee \neg Q(x_1, y_1, z))\end{aligned}$$

Exemplo

Seja $\varphi = (P \wedge \exists y S(y)) \wedge (\forall x Q(x) \vee R)$

$$\begin{aligned}\text{Scope}(\varphi) &= \text{Scope}(\text{Scope}(P \wedge \exists y S(y)) \wedge \text{Scope}(\forall x Q(x) \vee R)) \\ &= \text{Scope}(\exists y_0 \text{Scope}(P \wedge S(y_0)) \wedge \forall x_0 \text{Scope}(Q(x_0) \vee R)) \\ &= \text{Scope}(\exists y_0 (P \wedge S(y_0)) \wedge \forall x_0 (Q(x_0) \vee R)) \\ &= \exists y_1 \text{Scope}((P \wedge S(y_1)) \wedge \forall x_0 (Q(x_0) \vee R)) \\ &= \exists y_1 \forall x_1 \text{Scope}((P \wedge S(y_1)) \wedge (Q(x_1) \vee R)) \\ &= \exists y_1 \forall x_1 ((P \wedge S(y_1)) \wedge (Q(x_1) \vee R))\end{aligned}$$